19970728 026

Asilomar Conference, November'96

Efficient Design Methods of Optimal FIR Compaction Filters for M-channel FIR Subband Coders†

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

Ahmet Kirac and P. P. Vaidyanathan
California Institute of Technology
Pasadena, CA 91125
Fax : (818) 795 8649 Phone : (818) 395 4681
Email : ppvnath@sys.caltech.edu

Abstract

We propose algorithms for the design of FIR compaction filters, which find applications in FIR subband coders. The techniques produce compaction gains very close to that of optimal compaction filters, for any fixed filter order and input autocorrelation. The main theme of the paper is the design of multistage FIR compaction filters based on an iterated linear programming approach. The theory behind this is presented followed by design examples and comparisons. Also, a noniterative algorithm much faster than other iterative optimization techniques (e.g. linear programming) will be briefly mentioned. Further details of noniterative techniques will be presented elsewhere.

1 Introduction

We will describe some efficient methods to design FIR compaction filters. These filters find application in M-channel FIR orthonormal filter banks [11]. Because of this basic application, we will refer to "M-channel compaction filters", although there will be only one filter to work with. The theory and design details depend on M.

Compaction filters have attracted a great deal of attention due mainly to the recently discovered fact [9, 13] that they are the building blocks of optimal orthonormal (paraunitary) filter banks. This connection however is made for the case where the filters are allowed to be ideal. A number of authors considered the finite order (FIR) compaction filter design problem for the two-channel case [2, 1, 14, 10, 6, 8] and for the M-channel case [7].

It is possible to design optimal FIR compaction filters by using linear programming. However, when the filter orders are high, the computational complexity of the linear programming technique becomes very high. We propose an algorithm to design the filters in multiple stages resulting in efficiency in both design and implementation phases. We also briefly describe a noniterative design technique called the window method. Also mentioned is a new analytical method in the two-channel case. The details of these noniterative design methods will be presented elsewhere [4].

2 The FIR energy compaction problem

Let $H(z)$ be an FIR filter of order N. Consider Fig. 1 where the input $x(n)$ is a zero-mean WSS random process with the power spectral density $S_{xx}(e^{j\omega})$. The

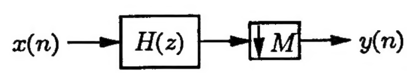

Figure 1: The FIR energy compaction filter.

output of the filter is decimated by M to produce $y(n)$. The optimum FIR energy compaction problem is to maximize the variance

$$\sigma_y^2 = \int_{-\pi}^{\pi} |H(e^{j\omega})|^2 S_{xx}(e^{j\omega}) \frac{d\omega}{2\pi} \tag{1}$$

of $y(n)$ subject to the Nyquist(M) condition [11] on $G(e^{j\omega}) = |H(e^{j\omega})|^2$. Let the impulse response of $G(e^{j\omega})$ be $g(n)$. Then, the Nyquist(M) condition is $g(Mn) = \delta(n)$. Notice that by definition $G(e^{j\omega}) \geq 0$. Define the **compaction gain** as

$$G_{comp}(M,N) = \frac{\sigma_y^2}{\sigma_x^2} = \frac{\int_{-\pi}^{\pi} |H(e^{j\omega})|^2 S_{xx}(e^{j\omega}) \frac{d\omega}{2\pi}}{\int_{-\pi}^{\pi} S_{xx}(e^{j\omega}) \frac{d\omega}{2\pi}} \tag{2}$$

where σ_x^2 is the variance of $x(n)$. The aim therefore is to maximize the compaction gain.

† Work supported in parts by Office of Naval Research grant N00014-93-1-0231, Tektronix, Inc., and Rockwell Intl.

DTIC QUALITY INSPECTED 4

As described in [4], the case where $N < M$ and the case where ideal filters are allowed are solved analytically. Our interest is therefore for the case where $M < N < \infty$.

3 Linear Programming

The use of linear programming method in compaction filter design was proposed in [6], and is reviewed next. Assume that the input process $x(n)$ is real with the autocorrelation sequence $r(n)$. The output variance can be written as

$$\sigma_y^2 = r(0) + 2\sum_{n=1}^{N} g(n)r(n) \tag{3}$$

Let $\mathbf{g}_d$ and $\mathbf{r}_d$ be the vectors formed by the nonzero components of $g(n)$ and $r(n)$ for $n = 1, \ldots N$. That is,

$$\begin{aligned} \mathbf{g}_d &= [g(1)\; g(2)\; \ldots\; g(M-1)\; g(M+1)\; \ldots\; g(N)]^T, \\ \mathbf{r}_d &= [r(1)\; r(2)\; \ldots\; r(M-1)\; r(M+1)\; \ldots\; r(N)]^T. \end{aligned}$$

Then (3) can be written as $\sigma_y^2 = r(0) + 2\mathbf{r}_d^T\mathbf{g}_d$. This incorporates the Nyquist(M) condition but not the nonnegativity constraint on $G(e^{j\omega})$. Let $\mathbf{c}_d(\omega) \triangleq [\cos(\omega)\; \cos(2\omega)\; \ldots\; \cos((M-1)\omega)\; \cos((M+1)\omega)\; \ldots\; \cos(N\omega)]^T$. Then $G(e^{j\omega}) = 1 + 2\mathbf{c}_d^T(\omega)\mathbf{g}_d$. Hence the problem is equivalent to the following:

$$\begin{gathered} \text{maximize } \mathbf{r}_d^T\mathbf{g}_d, \\ \text{subject to } \mathbf{c}_d^T(\omega)\mathbf{g}_d \geq -0.5,\ \forall\omega \in [0, \pi]. \end{gathered}$$

This type of problem is typically classified as semiinfinite linear programming [6]. By discretizing the frequency, one reduces this to a well known standard linear programming problem.

Example 1. Let the input process be AR(5) as in page 37 of [3] which is used to model speech signals. Let $M = 8$ and $N = 15$ and let us use $L = 32$ uniform frequencies, $\omega_k = k2\pi/L, k = 0, \ldots, L-1$. We obtain the frequency response $G(e^{j\omega})$ shown in Fig. 2 which is not nonnegative for all frequencies. In order to obtain a compaction filter, we must have $G(e^{j\omega}) \geq 0$. One way to guarantee this is to "lift" $G(e^{j\omega})$ by increasing $g(0)$ relative to the other coefficients (since $g(0)$ has to be 1, in effect we rescale $g(n)$ for $n \neq 0$ by a constant that is slightly less than 1). For this example, it turns out that this constant is about 0.8044. The resulting compaction gain is 3.7873. Optimum ideal compaction filter gives a gain of 4.9721.

In the next section we propose a new technique to overcome the difficulty without having to find the minimum of $G(e^{j\omega})$ which is necessary to decide on how much to lift it.

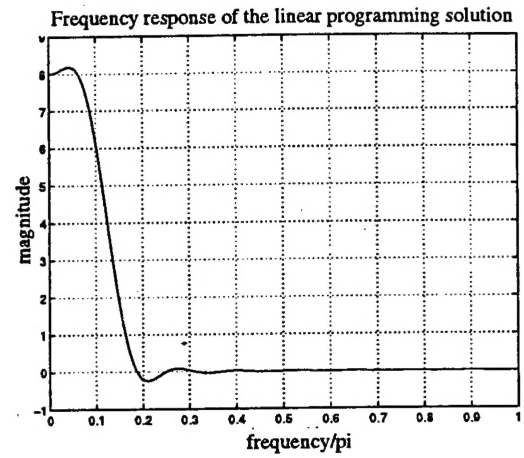

Figure 2: Linear programming solution, Ex. 1.

3.1 Windowing of the linear programming solution

Consider the periodical expansion $g_L(n)$ of the linear programming solution where L is the number of discrete uniform frequencies used in the design process. Assume that $L > 2N$. Linear programming assures that $G(e^{j\omega})$ is nonnegative at the uniform frequencies. Hence the Fourier series coefficients $G_L(k)$ of $g_L(n)$ are nonnegative. Now consider the product

$$w(n)g_L(n) \tag{4}$$

where $w(n)$ is a symmetric window of length $2K+1$ (where $K < L - N$, see Fig. 3). If $w(n)$ has nonnegative Fourier transform $W(e^{j\omega})$, the Fourier transform of the product is nonnegative as well. The reason follows from the fact that the Fourier transform of $w(n)g_L(n)$ is a weighted sum of shifted versions of $W(e^{j\omega})$, with nonnegative weights. For maximum compaction gain, the symmetric order of $w(n)$ is chosen to be maximum, namely $K = L - N - 1$. One can use a fixed window

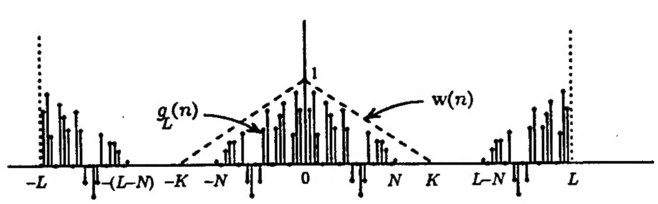

Figure 3: Windowing of the linear programming solution.

like a triangular window as depicted in the figure and get a satisfactory compaction gain. However one can

always optimize the window. The optimum $w(n)$ is the autocorrelation sequence of the maximal eigenfilter of the $K \times K$ Hermitian Toeplitz matrix formed by the product $r(n)g_L(n)$ [4]. Since the window length $2K+1$ is very high in linear programming designs, we suggest to use a triangular window rather than optimizing the window. The performance loss is negligibly small.

Example 2. Consider the previous example. Using a triangular window of order $K = L - N - 1 = 16$, we have the resulting compaction gain of 4.4967. This is significantly better than that of the previous "lifting" technique in Example 1. When we further optimize the window, we find that the compaction gain is 4.6769. If $L = 256$ however, we have the scaling constant 0.9986 in the lifting method with a corresponding compaction gain of 4.8471. If we use triangular window of order $L - N - 1 = 240$, we have the compaction of 4.8236 and if we optimize the window we have the compaction gain of 4.8539. Recall that the ideal compaction gain was 4.9721.

Example 3. Let the input be psd be as in Fig. 4 and let $N = 65$ and $M = 2$. In the same figure, we plot the magnitude square $|H(e^{j\omega})|^2$ of the compaction filter $H(z)$ designed by the linear programming method.

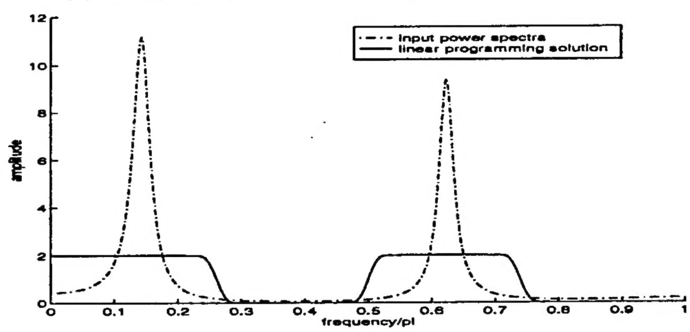

Figure 4: The psd of an AR(5) process, and the magnitude square of an optimal compaction filter designed by linear programming (N = 65, M = 2).

The number of frequencies used in the design process was $L = 512$. We have used triangular window of symmetric order $K = L - N - 1 = 446$ and found that the resulting compaction gain is 1.8698. If we optimize the window the compaction gain becomes 1.8744. If we "lift" $G(e^{j\omega})$, then we have the compaction gain of 1.8713. One can verify that the compaction gain of the ideal (infinite order) compaction filter is 1.8754.

4 Multistage FIR (IFIR) compaction filter design

Let $M = M_0M_1$ and consider Fig. 5(a). This can be redrawn as in Fig. 5(b). The equivalent filter is $H(z) = H_0(z)H_1(z^{M_0})$. We will first impose the Nyquist(M) condition only on $|H(e^{j\omega})|^2$. Later we will impose Nyquist conditions on individual filters that guarantee the Nyquist(M) property of $|H(e^{j\omega})|^2$. We will describe the details of how to find $H_1(z)$ for a fixed $H_0(z)$ and vice versa, in an iterative manner.

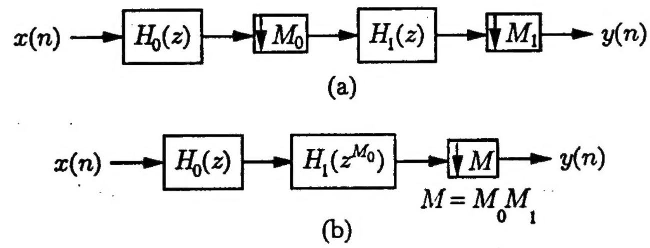

Figure 5: Multistage compaction filter design. (a) Basic configuration, (b) Equivalent system.

Let $G_0(e^{j\omega}) = |H_0(e^{j\omega})|^2$, $G_1(e^{j\omega}) = |H_1(e^{j\omega})|^2$, and $G(e^{j\omega}) = |H(e^{j\omega})|^2$ with impulse responses $g_0(n)$, $g_1(n)$, and $g(n)$ respectively. Denote the orders of $H_0(z)$, $H_1(z)$, and $H(z)$ by N_0, N_1, and N respectively. Hence we have $N = M_0N_1 + N_0$. Define

$$\mathbf{g_0} = [g_0(0)\ g_0(1)\ \ldots\ g_0(N_0)]^T,$$
$$\mathbf{g_1} = [g_1(0)\ g_1(1)\ \ldots\ g_1(N_1)]^T,$$
$$\mathbf{g} = [g(0)\ g(1)\ \ldots\ g(N)]^T.$$

Optimization of $H_1(z)$ for a given $H_0(z)$. We have $G(z) = G_0(z)G_1(z^{M_0})$. Let $\mathbf{G_0}$ be the $(2N+1) \times (2M_0N_1+1)$ convolution matrix formed by $g_0(n)$. Taking into account the symmetries and the fact that $G_1(z^{M_0})$ has nonzero components only for multiples of M_0, we can write $\mathbf{g} = \mathbf{A_0 g_1}$, where $\mathbf{A_0}$ is an $(N+1) \times (N_1+1)$ matrix that is obtained from $\mathbf{G_0}$. Now, the Nyquist(M) constraint requires that if we decimate $\mathbf{g}$ by M we should get $\mathbf{e_0} = [1\ 0\ \ldots\ 0]^T$. Let $\mathbf{B_0}$ denote the matrix that is obtained by taking every Mth row of $\mathbf{A_0}$. Then we should have $\mathbf{B_0 g_1} = \mathbf{e_0}$. To force the nonnegativity constraint on $G_1(e^{j\omega})$, let $\mathbf{c_0}(\omega) \triangleq [1\ 2\cos(\omega)\ 2\cos(2\omega)\ \ldots\ 2\cos(N_1\omega)]^T$. Then the constraint $G_1(e^{j\omega}) \geq 0$ becomes $\mathbf{c_0}^T(\omega)\mathbf{g_1} \geq 0$, $\forall\omega \in [0,\pi]$. If $\mathbf{r} = [r(0)\ 2r(1)\ \ldots\ 2r(N)]^T$, the objective is to maximize $\mathbf{r}^T\mathbf{g} = \mathbf{r}^T\mathbf{A_0 g_1}$. Hence we have reduced the problem to the following:

$$\text{maximize } \mathbf{r_0}^T\mathbf{g_1},$$
$$\text{subject to } \mathbf{B_0 g_1} = \mathbf{e_0}, \text{ and } \mathbf{c_0}^T(\omega)\mathbf{g_1} \geq 0,\ \forall\omega \in [0,\pi],$$

where $\mathbf{r_0} = \mathbf{A_0}^T\mathbf{r}$. Hence a standard linear programming algorithm can be applied, once a set of frequencies is chosen for the inequality constraint.

Optimization of $H_0(z)$ for a given $H_1(z)$. Similarly, one can reduce the problem of finding the best $H_0(z)$ for a given $H_1(z)$ to the following linear programming

problem:

$$\text{maximize } \mathbf{r_1}^T\mathbf{g_0},$$

subject to $\mathbf{B_1 g_0} = \mathbf{e_0}$, and $\mathbf{c_1}^T(\omega)\mathbf{g_0} \geq 0,\ \forall\omega \in [0,\pi]$,

where $\mathbf{c_1}(\omega) = [1\ 2\cos(\omega)\ 2\cos(2\omega)\ \ldots\ 2\cos(N_0\omega)]^T$, $\mathbf{r_1} = \mathbf{A_1}^T\mathbf{r}$. The $(N+1)\times(N_0+1)$ matrix $\mathbf{A_1}$ is obtained from the $(2N+1)\times(2N_0+1)$ convolution matrix formed by $g_1(n)$ by taking the symmetries into account and the matrix $\mathbf{B_1}$ is obtained by taking every Mth row of $\mathbf{A_1}$.

One can iterate between the above two optimization steps until there is no significant change in the compaction gain. The initial choice of $g_0(n)$ can significantly affect the resulting compaction gain. According to our design experience if $g_0(n)$ is chosen to be a triangular sequence, the compaction gain at the end of the iteration is very good. The filters $g_0(n)$ and $g_1(n)$ which result from the iteration should spectrally be factorized to identify $H_0(z)$ and $H_1(z)$. This step will be successful only if the solutions are such that $G_0(e^{j\omega}) \geq 0$ and $G_1(e^{j\omega}) \geq 0$ for all ω. If this is not the case, we can force it by use of windowing on $g_0(n)$ and $g_1(n)$ as described in Sec. 3.1 or by the "lifting" technique. If this is done then the product filter $G_0(z)G_1(z^{M_0})$ will not be exactly Nyquist(M). In the next subsection we show how to overcome this problem.

Example 4. Let us design IFIR compaction filters for the pair $(M,N) = (36,65)$, and for the input process whose psd was given in Fig. 4. Let $M_0 = 9$ and $M_1 = 4$, and let $N_0 = 11$ so that $N_1 = 6$. The number of frequencies used in the designs is $L = 1024$. Starting with a triangular sequence for $g_0(n)$, the algorithm converges in a few steps. We windowed the resulting solutions $g_0(n)$ and $g_1(n)$ with triangular windows of symmetric orders $L-N_0-1$ and $L-N_1-1$ respectively. The final product filter was not exactly Nyquist(M) because it was found that $g(36) \simeq -0.0018 \neq 0$. The final compaction gain was 5.1444. If we design a compaction filter of order 18 directly (i.e., not using IFIR technique), the compaction gain is 4.4225. This corresponds to a compaction filter with the same number of active multipliers, namely 19. If we design a compaction filter of order 65 directly (66 active multipliers), then the resulting compaction gain is 7.2337.

4.1 A Particular IFIR configuration

In Fig. 5, if $G_0(z)$ is Nyquist(M_0) and $G_1(z)$ is Nyquist(M_1), it can be verified that $G(z)$ given by $G_0(z)G_1(z^{M_0})$ is Nyquist(M). Now, let us fix $H_0(z)$ to be a valid compaction filter for the pair (N_0, M_0). Referring to Fig. 6(a), the best $H_1(z)$ is the optimum compaction filter for (N_1, M_1), and for the input $x_0(n)$ which has the psd $S_{x_0x_0}(z) = \Big(G_0(z)S_{xx}(z)\Big)\Big|_{\downarrow M_0}$. Similarly, if $H_1(z)$ is a fixed compaction filter for the

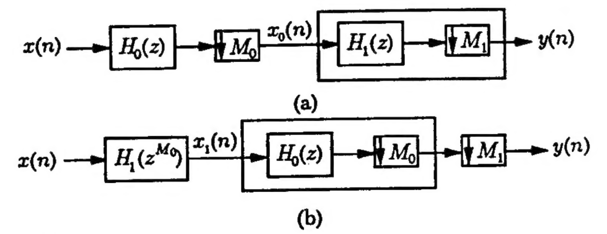

Figure 6: Special IFIR design configuration.

pair (N_1, M_1), then we can redraw the configuration as in Fig. 6(b). The best $H_0(z)$ is the optimum compaction filter for (N_0, M_0), and for the input $x_1(n)$ which has the psd $S_{x_1x_1}(z) = G_1(z^{M_0})S_{xx}(z)$. One can design the compaction filters $H_0(z)$ and $H_1(z)$ iteratively using any of the known techniques. Hence, one can use the linear programming technique as well as any other technique like the noniterative methods to be mentioned in the next section. Also note that if the ideal filters are allowed, this multistage configuration has no loss of generality as shown in [5].

Example 5. Let the setup be the same as in the previous example. We have designed the compaction filters $H_0(z)$ and $H_1(z)$ iteratively using the standard linear programming procedure as in Example 3. We have started with $H_1(z) = 1$. The first compaction filter $H_0(z)$ is therefore the optimal compaction filter for the pair $(M_0, N_0) = (9,11)$ for the original autocorrelation sequence. We have windowed the final product filters as we did in Example 3 to guarantee the nonnegativity. The resulting overall compaction gain is 4.9432. This is slightly smaller than the overall compaction gain 5.1444 in Example 3. However, the resulting overall filter here is exactly Nyquist(M) unlike the case of Example 4.

4.2 Noniterative techniques

In [4] we propose two noniterative methods for the design of FIR compaction filters. One is for any number of channels while the other is for the special two-channel case. The first one is called the **window method.** Although it is suboptimal, the window method is applicable for any process including complex ones. It has finite number of elementary steps and the resulting compaction gains are very close to the optimal ones especially for high filter orders. The second method is called the **analytical method.** It finds the optimal solution, but it is applicable for a restricted class of random processes. Because of its relevance and similarities to the linear programming method, we will

describe the window method briefly. The details of both methods are presented in [4].

Window method. The main idea is to write the impulse response of $G(e^{j\omega}) = |H(e^{j\omega})|^2$ in the form:

$$g(n) = g_L(n)w(n) \tag{5}$$

where $g_L(n)$ is periodic(L), and $w(n)$ has nonnegative Fourier transform. The method takes the window to be a triangular one, and finds the optimum $g_L(n)$. By fixing this $g_L(n)$, the window $w(n)$ is then optimized. Optimizing $g_L(n)$ for a fixed $w(n)$ is done by comparing the psd at M alias frequencies for each $\omega_k = k2\pi/L$ and assigning values to $G_L(k)$ accordingly. Here $G_L(k)$ is the Fourier series coefficients of $g_L(n)$. This can be considered as finite version of the algorithm in [12]. Optimizing $w(n)$ for a fixed $g_L(n)$ is an extremal eigenvector problem [4].

Relation to linear programming. In linear programming we find a sequence whose Fourier transform is nonnegative at some prescribed set of L frequencies. If these are chosen to be L uniform frequencies $\omega_k = k2\pi/L, k = 0, \ldots, L-1$, then we can associate a periodic sequence $g_L(n)$ whose Fourier series coefficients are nonnegative. If we window this with $w(n)$ whose Fourier transform is nonnegative as in Sec. 3.1, we guarantee the nonnegativity of $G(e^{j\omega})$. Hence in principle, we have the same form for $g(n)$ as in (5). There are some basic differences however: In linear programming $g_L(n)$ is automatically restricted to be of finite length. That is, it is guaranteed that $g_L(n) = 0$ for $N < |n| < L-N$. This is, in general, not true in the window method. The order of $w(n)$ should be the same as that of $g(n)$ in the window method while this is not necessary in the linear programming method. In the special case where $L = 2N$, the two methods become the same! Hence the window method becomes an efficient way of solving a linear programming problem. In [4] we give a detailed comparison of the two methods.

5 Remarks and Conclusions

We have discussed some efficient design methods for FIR compaction filters. We first proposed a simple way to guarantee the nonnegativity of the linear programming solutions. Then we have considered multistage extensions. These offer reduction in both design and implementation complexity. We have also briefly described a noniterative technique called the window method and discussed its relevance to the linear programming technique. When the number of frequencies in the linear programming is moderate, we have seen that the "lifting" technique to assure nonnegativity of $G(e^{j\omega})$ resulted in significant loss. In these cases optimization of the window in the windowing technique was the best. When the number of frequencies is high however, then either "lifting" or the use of a triangular window resulted in very little loss in compaction gain. Finally we noted that when $L = 2N$, the linear programming and the window methods become the same. Since the window method is much faster, it should be preferred to the linear programming technique if L is chosen to be moderate (e.g., not much larger than $2N$). We conclude the paper by referring to an important observation made in [4] that states that if the filter order is relatively high, then the linear programming technique can be avoided altogether because the choice $L = 2N$ yields very good window-based design.

References

[1] H. Caglar, Y. Liu, and A. N. Akansu. Statistically optimized PR-QMF design. In *SPIE, Visual Comm. and Image Proc. '91: Visual Comm.*, volume 1605, pages 86–94, 1991.

[2] P. Delsarte, B. Macq, and D. T. M. Slock. Signal-adapted multiresolution transform for image coding. *IEEE Trans. on Inform. Theory*, IT-38(2):897–904, March 1992.

[3] N. S. Jayant and P. Noll. *Digital coding of waveforms.* Englewood Cliffs, NJ: Prentice Hall, 1984.

[4] A. Kirac and P. P. Vaidyanathan. *Theory and design of optimum FIR compaction filters.* Caltech, technical report, Sep. 1996.

[5] Y.-P. Lin and P. P. Vaidyanathan. Considerations in the design of optimum compaction filters for subband coders. In *Eusipco*. Trieste, Italy, 1996.

[6] P. Moulin. A new look at signal-adapted QMF bank design. In *Proc. of the IEEE ICASSP-95*, pages 1312–1315, May 1995.

[7] P. Moulin, M. Anitescu, K. Kortanek, and F. Potra. Design of signal-adapted FIR paraunitary filter banks. In *Proc. of the IEEE ICASSP-96*, volume 3, pages 1519-1522. Atlanta, May 1996.

[8] J.-C. Pesquet and P. L. Combettes. Wavelet synthesis by alternating projections. *IEEE Trans. on Signal Proc.*, SP-44(3):728–732, March 1996.

[9] M. Unser. On the optimality of ideal filters for pyramid and wavelet signal approximation. *IEEE Trans. on Signal Proc.*, SP-41(12):3591–3596, Dec. 1993.

[10] B. Usevitch and M. T. Orchard. Smooth wavelets, transform coding, and markov-1 processes. In *Proc. of the IEEE ISCAS-93*, pages 527–530, May 1993.

[11] P. P. Vaidyanathan. *Multirate systems and filter banks.* Englewood Cliffs, NJ: Prentice Hall, 1993.

[12] P. P. Vaidyanathan. *Optimal orthonormal filter banks.* Caltech, technical report, Sep. 1995.

[13] P. P. Vaidyanathan. Theory of optimal orthonormal filter banks. In *Proc. of the IEEE ICASSP-96*, volume 3, pages 1487–1490. Atlanta, May 1996.

[14] L. Vandendorpe. CQF filter banks matched to signal statistics. *Signal Proc.*, 29:237–249, 1992.